Dear Team Members,

I hope you are doing GREAT!

Fashion MNIST is an alternative to MNIST and another default dataset that we can access through Keras.

Fashion MNIST is a dataset of 60,000 28x28 grayscale images of 10 fashion categories, along with a test set of 10,000 images. This dataset can be used as a drop-in replacement for MNIST. The class labels are:

1. T-shirt/top
2. Trouser
3. Pullover
4. Dress
5. Coat
6. Sandal
7. Shirt
8. Sneaker
9. Bag
10. Ankle boot

The dataset is a bit more challenging compared to MNIST, and the achieved accuracy is normally less than what we can get from the classical MNIST.

This is where we will start importing and loading our dataset:



Here are what we need to do:

1. What are the dimensions of train\_images, train\_labels, test\_images, and test\_labels?
2. What are the lengths of train\_labels and test\_labels?
3. Please show some of train and test labels.
4. Please show the digital content of image index 5 in the training dataset.
5. Please plot the image of the index 5 in the training dataset.
6. What is the label for the index 5 in the train\_label and looking up in the above list, what does it mean?
7. Please show the digital content of image index 500 in the testing dataset.
8. Please plot the image of the index 500 in the testing dataset.
9. What is the label for the index 500 in the test\_label and looking up in the above list, what does it mean?
10. Please import models and layers from the keras library.
11. Define a sequential model and call it myNetwork.
12. Reshape the images from 28x28 to one column with 784 neurons (flattening) (use 2 methods, one from the book and one from the 20-minute video).
13. Also, please normalize the image by dividing the image by 255 (use 2 methods, one from the book and one from the 20-minute video).
14. Add one hidden layer that has 512 neurons, using ‘relu’ activation function.
15. Add another hidden layer that has 128 neurons, using ‘relu’ activation function.
16. Add the last layer as a 10-neuron dense layer that uses the ‘softmax’ as the activation function. Why we use softmax for the last layer? How does it work under the hood?
17. Use the following two settings for the compiler and run them separately and see what the differences are.

* Optimizer 🡺 adam, loss🡺 'sparse\_categorical\_crossentropy', metrics🡺[‘accuracy’]
* Optimizer 🡺 ramsprop, loss🡺 'categorical\_crossentropy', metrics🡺[‘accuracy’]

1. Now after the compilation, please try to find the pattern using the fit command. The epochs need to be 10 for this example.
2. How do you compare the fashion\_MNIST with what we learned in the class using the MNIST? What can we infer from the differences in the accuracy? What could be the reasons for that?
3. Use the evaluate to calculate the achieved accuracy and loss over the test images and labels. Do we have overfitting?

**Deliverable:**

Please note that we need **two separate components** to submit for the assignment 01.

**Part A:**

An ipynb file that contains all the code and please mention the number that belongs to the question, e.g.

#01 from keras.datasets import fashion\_mnist

I strongly suggest using the Google Colab, but please choose your favorite platform. Every question has 0.25 and it provides five out of total ten marks for the assignment.

**Part B:**

Please record and upload a video to YouTube, describing all you have done in 5 to 10 video recorded clip (That 20-minute video could be a sample). If you go beyond 10 minutes, it is fine, and the mentioned time range is to show how the scale of the assignment could be. Then submit a notepad text document that contains the link to the uploaded YouTube video.

**Summary:**

In short, what we need to submit are:

1. An ipynb file that contains all the code, and they need to be numbered. (5 marks)
2. A text file that contains the YouTube link of your recorded video. (5 marks)

Wish you only the best, Dear Team Members!

Reza